CSE 5337/7337: Information Retrieval and Web Search

Spring 2018, Project 2: Query engine implementation (100 points)

Due: Tuesday May 8, 6:00 pm

Demonstrations are ***encouraged***, but not required.

Deliverables:

1. Complete code in a compressed archive (zip, tgz, etc)
2. A readme file with complete description of used software, installation, compilation and execution instructions to allow me to install and run your program if needed.

*If a demonstration is not scheduled, then I must be able to install and run your program to ensure it works as you described, so test your directions!*

1. A document with the results for the tasks below.

Task:

**Develop a simplified query engine.**Test your data only on the data in:

<https://lyle.smu.edu/~fmoore> aka <https://s2.smu.edu/~fmoore>

1. Use the web crawler you built in Project 1 that crawled a limited space, looking for text, html files and php files. Everyone will need to save the text from the <TITLE> tag. You may need to modify how you saved the information from the documents that you traversed to support the query engine. Describe in detail what was changed to support the second half of the project. [10/10 points]
2. For the purpose of this project, you may assume a maximum of 60 documents. You will need to cluster the documents into 5 leader/follower sets. [30/25 points]  
   a) How did you determine the leaders?  
   b) List the leader-id, document-id, score for each pair.
3. The user will be able to enter multiple queries, consisting of one or more query words separated by space. The single word query “stop” will cause your program to stop. [10/10 points]   
   a) What happens if a user enters a word that is not in the dictionary?   
   b) What happens if a user enters a stop word?  
   c) A set of queries will be provided.
4. Implement the cosine similarity of the query against each leader document. [40/30 points]   
   a) What document/query weighting scheme did you implement?   
   b) If any of the query words appear in the <title> of any selected document, add 0.25 to its score.  
   c) Display the resulting score, document URL, and document title in descending numerical order for the top K=6 results.
5. Explain why you believe the results are correct. [10/10 points]

**GRADUATE STUDENTS:**

1. Include in the display, the first 20 words of the document (this can be the stemmed version). [0/5 points]
2. If less than K/2 documents are returned for a query, rerun the query using thesaurus expansion. A list of words, along with 1 – 3 synonyms will be provided. [0/10 points]

You must show the results of these queries (tentative):  
1. moore smu  
2. Bob Ewell where Scout  
3. three year story  
4. Atticus to defend Maycomb  
5. hocuspocus thisworks

You may certainly include additional queries that reflect your testing.

A small thesaurus is provided here. It is used when a query returns less than 3 documents. When that happens, internally expand the query using thesaurus expansion on each term and then execute the revised query.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| word | alternates | | | |
| beautiful | nice | fancy |  |  |
| chapter | chpt |  |  |  |
| chpt | chapter |  |  |  |
| responsible | owner | accountable |  |  |
| freemanmoore | freeman | moore |  |  |
| dept | department |  |  |  |
| brown | beige | tan | auburn |  |
| tues | Tuesday |  |  |  |
| sole | owner | single | shoe | boot |
| homework | hmwk | home | work |  |
| novel | book | unique |  |  |
| computer | cse |  |  |  |
| story | novel | book |  |  |
| hocuspocus | magic | abracadabra |  |  |
| thisworks | this | work |  |  |

(end)